均等分区问题模型、算法及应用

孔云峰 [1]

1 河南大学黄河中下游数字地理技术教育部重点实验室，河南 开封，475000

摘要：分区问题广泛应用于地理、经济、政治、商业、公共服务等领域。均等分区问题是其中一类问题，通常要求分区人口数量或任务量均等、几何形状紧凑和空间连续，应用于选区、销售区和巡逻区的划分。本文针对均等分区问题提出了一个混合整型线性规划模型，并设计了一个基于迭代局部搜索(ILS)的混合算法。该算法从三个方面扩展 ILS：群解搜索、VND 搜索及 SPP 模型改进。选择 5 个区域对模型和算法进行测试，结果表明：数学模型能够求解空间单元数量为 324 的案例；混合算法优化性能优异，鲁棒性强，计算效率较高。所提出的均等分区问题适用于政治分区等经典问题，在新冠疫情应急服务等领域也具有应用潜力。

关键词：均等分区问题；混合整型线性规划；混合算法；应急服务

分区问题是将特定地理空间划分为若干区域，满足一定的约束条件，并达到最优的分区目标，广泛应用于地理、经济、政治、环境、生态、农业、城市、商业、公共服务、地图制图、空间统计等领域。分区问题类型繁多，涉及分区数量、单个或多个分区指标、静态或时间序列指标、是否要求分区空间连续等。根据分区遵循的基本原则，分区问题可分以三大类：(1)均质性分区，即分区内差异最小化和分区间差异最大化，如自然地理、气候和农业区划；(2)功能性分区，即分区内部交互作用最大化(区内凝聚)和分区间交互作用最小化(区间分离)，如经济功能区和城市功能区划分；(3)均衡性分区，如分区人口数量或其他属性均等的选区、警察巡逻区和销售区划分，供应与需求均衡的学区和医疗服务区划分。自 1960s 以来，学术界一直在探索各类分区问题求解方法，相关算法研究文献十分丰富。然而，由于分区问题涉及领域广，问题类型多，分区原则、目标函数和约束条件各有差异，求解分区问题的模型和算法仍在持续探索之中。

选区、销售区、警察巡逻区划分等问题是一类常见的均等分区问题(EDP)。此类问题均要求分区空间连续、分区人口数量或任务量均等、几何形状紧凑和基本空间单元完整(不被分割)。选区划分问题是遵循法律强制要求，基于人口统计基本单元将一个区域划分为若干选区，在选区中进行政治选举。选区划分通常优先考虑选区人口均等、空间连续和几何形状紧凑原则，从而避免选区划分的不公平性；其他准测(如与行政区吻合度，种族、社会、经济、政治属性均衡分布等)并未形成共识[1]。销售区划分问题是企业将其市场区域划分为若干分区，指派一个雇员或一个团队负责一个区域进行产品销售或客户服务，从而合理、高效地进行企业运作管理。销售区划分首先考虑雇员工作量均等，其次是分区紧凑和空间连续，从而减少旅行开支。类似地，警察巡逻区划分要求分区的巡逻任务量均等，并考虑分区紧凑和空间连续，从而提高巡逻效率。

现实生活中的某些设施服务区设计也要求服务区均等、紧凑和连续。例如，公安警务室设计。若每个警务室服务范围人口均等、形状紧凑和空间连续，则使片警管辖范围明确、任务量均衡且居民办理相关事务的旅行距离短。出现重大疫情实施封闭小区措施时，为保证居民生活物资配送，也可以将实施封闭区域划分为若干服务区。均等、紧凑和连续的服务区设计，能够使得居民生活物资保障人员的任务量均衡，上门旅行距离较短，且服务范围明确。新冠病毒全民核酸检测亦需要分区服务规划。例如，疾控部门

收稿日期：2021-00-00
项目资助：国家自然科学基金项目(41871307)
通讯作者：孔云峰，博士，教授，主要从事空间分析、空间优化等研究。yfkong@henu.edu.cn

派出多个工作组对某一区域进行检测，每个工作组设置一个服务站，计划在最短时间内完成检测。每个工作组的服务区要求任务均等、服务范围紧凑且连续。服务区紧凑可使每个服务站服务的居民空间集中度高、到达服务站步行距离最短，也能避免与居住较远的居民产生聚集。

求解分区问题的主流算法包括精确算法和启发式算法[2]。因分区问题是 NP-Hard 问题[3]，数学模型求解能力有限。文献[4]为均质性分区问题构建了三个数学模型，然而，针对 49 个单元的案例，3h 的计算时间仍难于获得最优解。文献[5]构造了一个政治分区问题数学模型，需要 6s~53,632s 求解 25~36 个空间单元的案例。为高效求解，现有模型多将问题简化，如不考虑空间连续、不考虑紧凑目标或者将单元分割[6-10]。将集合划分问题(set-partitioning problem, SPP)模型用于分区问题，要先构造大量分区方案，通过模型选择最优的分区组合[11-13]；虽然集合划分问题模型相对容易求解，但求解结果取决于候选分区的数量和质量。

分区问题的元启发算法很多，如基于邻域搜索的各种贪婪 (Greedy)、模拟退火(SA)、禁忌搜索(TS)和动态门槛法(OBA)算法[14-19]，基于自然进化的遗传(GA)、演化(EA)、分散搜索(SS)、智能蜂群(ABC)算法[22-28]。然而，优化性能较好的元启发算法，算法机制设计较为复杂，计算时间也偏长，如 PEAR 算法[28]需要设置 19 个算法参数，在设置 13.1 万处理器的超级计算机上运行长达 3 小时的时间。另外，区域生长方法[29][30]和计算几何方法[31-33]思路简单，但优化性能有限。总体上，现有分区问题算法往往难以兼顾算法设计难度、优化性能和计算效率。

本文针对均等分区问题，构造了一个混合整型线性规划(MILP)模型，满足分区均等、紧和和连续要求，设计了一个基于迭代局部搜索(ILS)的混合算法。模型和算法是、的有效性在五个案例区得到了验证。

1 问题定义及数学模型

对于一个地理区域，包含 n 个空间单元。令集合 V={1,2…n}表示 n 个空间单元，每一个空间单元 i 具有属性 p_i，表示空间单元 i 人口数量或任务量；变量 c_{ij} 代表空间单元 i 和 j 是否邻接(有公共边)；集合 N_i 表示单元 i 的相邻单元，即 $N_i=\{j,\forall j\in V,c_{ij}=1\}$；变量 d_{ij} 表示空间单元 i 与学校单元 j 之间的距离。本文将均等分区问题定义为：将空间单元集合 V 划分为 K 个子集(即 K 个分区)，记为集合 R={r_1,r_2…r_K}，要求基本空间单元完整、分区空间连续、分区人口或任务量均等、分区几何形状紧凑等。这一定义适用于选区、销售区、巡逻区等分区问题。

空间连续和紧凑的表达方式是构建分区问题数学模型的关键。将地理空间表达为一个网络图 G={V，E}，即将空间单元作为结点，相邻空间单元($c_{ij}=1$)相连接作为网络边，可采用“网络流”概念表达分区空间连续性[4] [34]。在一个分区内，每个单元产生一个单位的流量，通过网络边流动，最终汇入到分区内某一汇集单元。汇集单元不产生流量，但汇入最高达 n-K 单位的流量。产流和汇流单元所构成的区域是一个空间连续的区域。令集合$N_i=\{j|c_{ij}=1\}$*表示*与单元 i 相邻的单元集合。文献[4]引入决策变量 W_{ik} 表示单元 i 是否分区 k 的汇流单元，决策变量 Y_{ik} 表示单元 i 是否属于分区 k，整型变量 f_{ijk} 表示单元 i、j 间在分区 k 中的流量，可构建线性不等式约束保证分区空间连续。

分区几何紧凑度指数广泛应用于分区几何形状评价。有多种几何紧凑度定义，然而每种定义具有一定的优势和局限[36]。常见几何紧凑度指标定义划分为两大类：第一类从平面几何角度定义，如$\sum_{k\in R}4A_k/\pi D_k^2$ / K(A_k、D_k 为分区 k 的面积和周长)；第二类从空间可达性的角度定义，如$\sum_{i\in r_k}d_{ic_k}$(c_k 为区域 k 的中心单元)。本文采用第二类紧凑度定义，即从每个单元到达所在分区中心单元的总距离作为紧凑度目标函数，总距离越短分区越紧凑。

将紧凑度作为目标，将分区均等和空间连续作为约束条件，构建如下整型线性规划模型：

$$\text{Min. } \sum_{i\in V}\sum_{k\in V}p_i d_{ik}x_{ik} \,/\, \sum_{i\in V}p_i \qquad (1)$$

$$\text{S.T. } \sum_{k\in V} x_{ik} = 1, \forall i \in V \tag{2}$$
$$(1-\varepsilon)\overline{Q}y_k \le \sum_{i\in V} p_i x_{ik} \le (1+\varepsilon)\overline{Q}y_k, \forall k \in V \tag{3}$$
$$\sum_{k\in V} y_k = K \tag{4}$$
$$f_{ijk} \le (n-K)x_{ik}, \forall i \in V, j \in N_i, k \in V \tag{5}$$
$$f_{ijk} \le (n-K)x_{jk}, \forall i \in V, j \in N_i, k \in V \tag{6}$$
$$\sum_{j\in N_i} f_{ijk} - \sum_{j\in N_i} f_{jik} \ge x_{ik}, \forall i \in V, k \in V, i \ne k \tag{7}$$
$$x_{ik} = \{0,1\}, \forall i \in V, k \in V \tag{9}$$
$$y_k = \{0,1\}, \forall k \in V \tag{10}$$
$$f_{ijk} \ge 0, \forall i \in V, j \in N_i, k \in V \tag{11}$$

模型中，目标函数(1)是紧凑度目标函数，分子部分表示所有单元与其分区中心的加权距离，分母部分表示地理区域内的总人口或总任务数量，商为平均距离。对于人口均等分区，目标函数值表示区域内所有人到达其中心点的平均距离。约束条件(2)保证每个单元必需划分在一个分区中。约束条件(3)要求分区在一定误差内均等，式中$\overline{Q} = \sum_{i\in V} p_i/K$ 为人口或任务量平均值，ε为限差，根据实际需求取值，如 1%、5%或其他值。约束条件(4)要求每个分区数量为 K。约束(5)-(7)是分区空间连续约束条件，条件(5)和(6)确保属于同一个分区的两个相邻单元间才可能产生流；若单元 i 不是汇集单元(中心单元)，约束(7)保证该单元至少产生一个单位的流。约束(5)-(7)将使每一个分区中的产流汇集到中心单元。条件(9)-(11)定义决策变量。

通过事先确定分区中心(流汇)单元可将模型简化。给定集合 $S=\{s_1,s_2 \dots s_K\}(s_k \in r_k, \forall k \in R)$表示 K 个分区的中心单元。移除决策变量y_k，以上模型可简化如下：

$$\text{Min.} \sum_{i\in V} \sum_{k\in S} p_i d_{is_k} x_{is_k} / \sum_{i\in V} p_i \tag{12}$$
$$\text{S.T. } \sum_{k\in S} x_{ik} = 1, \forall i \in V \tag{13}$$
$$(1-\varepsilon)\overline{Q} \le \sum_{i\in V} p_i x_{is_k} \le (1+\varepsilon)\overline{Q}, \forall k \in S \tag{14}$$
$$f_{ijk} \le (n-K)x_{ik}, \forall i \in V, j \in N_i, k \in S \tag{15}$$
$$f_{ijk} \le (n-K)x_{jk}, \forall i \in V, j \in N_i, k \in S \tag{16}$$
$$\sum_{j\in N_i} f_{ijk} - \sum_{j\in N_i} f_{jik} \ge x_{ik}, \forall i \in V, k \in S, i \ne k \tag{17}$$
$$x_{ik} = \{0,1\}, \forall i \in V, k \in S \tag{18}$$
$$f_{ijk} \ge 0, \forall i \in V, j \in N_i, k \in S \tag{19}$$

本质上，模型(1)-(11)相当于增加了空间连续约束的 Location-Allocation 问题模型；而简化模型(12)-(19)相当于增加了空间连续约束的 Allocation 问题。与原问题相比，简化问题不需要确定分区中心，且决策变量数量大幅减少。也应带注意到，容量约束 Allocation 问题仍是 NP-Hard 问题，简化模型仍具有很高的计算复杂度。文献[37]利用 K-medoids 算法确定分区中心，再使用与(12)-(19)接近的模型进行分区，计算效率较高。

2 算法设计

2.1 基本算法

选择迭代局部搜索(ILS)算法作为求解均等分区问题的算法框架。ILS 算法思路简单、易于实现，对于离散优化问题行之有效[38]。该算法从一个初始解开始，迭代地进行扰动和局部搜索。局部搜索容易陷入局部最优，对当前位置的扰动能够使算法脱离局部最优。ILS 算法中的核心部件包括始解生成、局部搜索和当前解扰动，示意如下：

Algorithm 1: Iterated local search

1. s_0 = GenerateInitialSolution();
2. s = LocalSearch(s_0);
3. Repeat until ter. condition met:
4. s' = Perturbation(s);
5. s^* = LocalSearch(s');

6. If $f(s^*) < f(s)$: $s = s^*$;

7. Output s.

有多个方法产生初始解，例如随机种子生长算法和加权 K-medoids 算法。随机种子生长算法原理如下：随机选择 K 个空间单元作为分区中心，按距离中心最近原则将所有单元指派给分区中心，从而形成初始分区。加权 K-medoids 算法[37]是一个加权聚类算法，随机选择 K 个空间单元作为分区中心，迭代进行加权指派和中心点更新，直到满足终止条件。这两种算法均不保证分区连续要求和分区均等要求。一般来说，两种算法获得分区结果接近空间连续，仅仅作少量的单元调整即可修复分区为连续。分区均等条件得不到满足时，在后续搜索中逐步改进，直到满足要求。通常的做法是在目标函数中加入非均等惩罚项，随惩罚项数值降低直至为零，分区均等条件得到满足。

ILS 局部搜索采用分区边界单元移动的方法[39]，包括每次移动一个单元的一单元移动算子、每次移动两个单元的二单元移动算子，以及每次移动三个单元的三单元移动算子。考虑到每次移动三个单元的计算复杂度很高，本文仅采用前两个算子。

对于空间均等分区，扰动方法很多。例如，破环若干分区、破环若干连续分区、破环一个连续区域、破环一定比例的边界单元，然后进行修复。也可以直接将若干边界单元移动到相邻分区中，之后根据需要进行连续性修复。

2.2 算法扩展

为提升优化性能，对 ILS 算法进行三方面的扩展。首先，改造单解 ILS 算法为群解 ILS 算法；其次，在 ILS 迭代循环中，使用变邻域下降(VND)搜索取代简单的局部搜索；第三，记录 ILS 搜索过程中产生的分区，ILS 搜索完成后使用 SPP 模型，进一步选择更优的方案。扩展后的算法是一个混合算法，流程如下：

Algorithm 2: ILS-based hybrid algorithm

参数：群大小（*psize*），破环强度(*strength*)，连续未更新最好解循环数(*mloops*), SPP 模型求解时间（*t*).

1. P = GenerateInitialSolutions(*psize*);
2. *pool*=*null*;
3. s_{best}=Best(P);
4. *notImpr*=0;
5. While *notImpr* < *mloops*:
6. Select a solution s from P randomly;
7. s'=Perturbation (s, *strength*);
8. s''=VNDsearch(s', *lsops*);
9. s^*=updateCenetrs(s'')
10. If $f(s^*)<f(s_{best})$: $s_{best}=s^*$, *notImpr=0;*
11. else: *notImpr*+=1*;*
12. *pool*=UpdateAreaPool(*pool*, s^*);
13. P =UpdatePopulation (P, s^*);
14. s=SetPartitioning (*pool, t*);
15. Output s.

与基于单解的搜索算法相比，本文算法维护一组解。首先，算法步骤(1)生成一组初始解；其次，在每一次迭代，从群解中随机选择一个解进行搜索(步骤 6)；第三，搜索完成后，使用新解更新群解(步骤 13)。群解更新中，优先考虑解的目标值，其次考虑解的差异程度，保持解之间有一定的差异。基于群解的 ILS 算法维护一组具有差异度的精英解，扩大了解空间搜索范围，有利于改进求解质量；同时，算法收敛速度通常会变慢，计算时间有一定的增加。

ILS 中使用 VND 搜索能够使每轮搜索达到局部最优。VND 方法反复使用邻域搜索算子进行搜索，直到所有算子均不能提升当前解，从而获得局部最优解。ILS 中频繁使用扰动方法，若当前解不是局部最优，下轮搜索中很容易被破坏，减少了到达局部最优的机会。因此，VND 搜索常用于 ILS 算法。

算法步骤(13)使用了 SPP 模型。SPP 模型求解时需要先构造大量分区，再通过模型选择最优的分区组合。ILS 搜索过程中，会产生大量的分区。所有的分区构成集合 Ω。每一个分区 i 具有目标值 O_i 和单元集合 U_i。SPP 模型从集合 Ω 中选择一个数量为 K 的子集，覆盖所有地理单元集合 V，并使目标值最小。定义 Ω_j 为 Ω 的子集，$\Omega_j = \{i | i \in \Omega, j \in U_i\}$，是一个包含空间单元 j 的分区的集合，定义决策变量 x_i 表明候选服务分区 i 是否被选中。SPP 模型如

下：

$$\text{Min.} \sum_{i\in\Omega} O_i x_i \quad (20)$$

$$\text{S.T. } \sum_{i\in\Omega_j} x_i = 1, \forall j \in V \quad (21)$$

$$\sum_{i\in\Omega} x_i = K \quad (22)$$

$$x_i = \{0,1\}, \forall i \in \Omega \quad (23)$$

3 实验验证

3.1 实验设计

收集5个区域的地理数据进行均等分区实验。案例区域范围小到一个住宅小区，大到省级行政区；空间单元个数从 28 到 2145；既有城市也有农村地区。案例区形状、空间单元及单元属性值分布如图 1 所示，自上而下从左到右区域名称简称为 HD、ZY、GY、GY2 和 HN。主要指标统计见表 1。

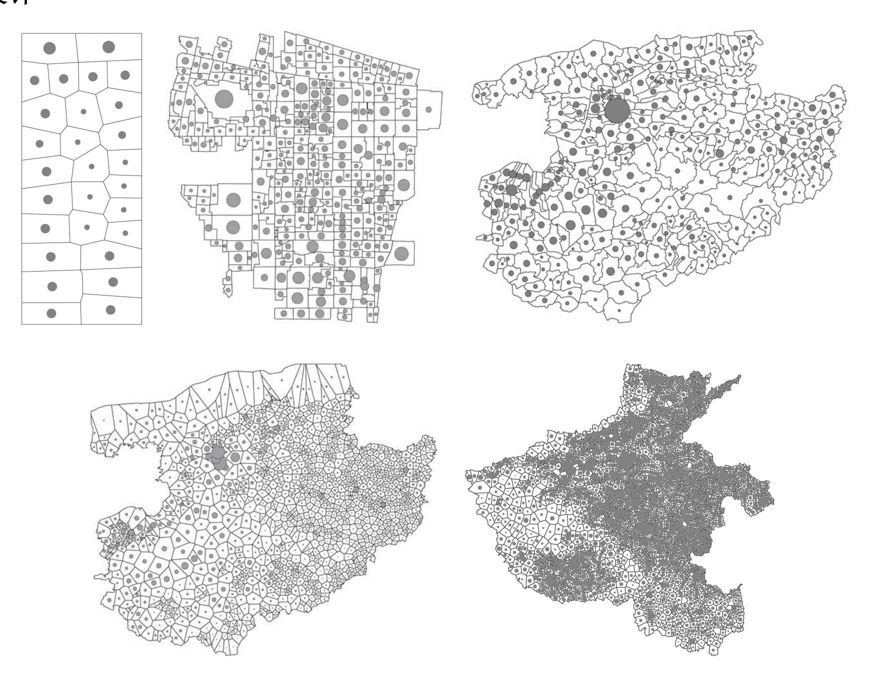


图 1 实验区示意图

Figure 1 The study areas

表 1 实验区主要特征

Table 1 Main features of the study areas

案例区域	HD	GY	ZY	GY2	HN
地理范围	居住小区	县级市	市辖区	县级市	省
空间单元	楼栋	行政村	小区	自然村	乡镇街道
空间单元数量	28	297	324	1,276	2,145
单元属性值之和	2,420	36,824	3,873	819,812	96,918,036

对五个区域进行均等化分。考虑到地理空间不可能严格均分，允许分区有一定的误差，即模型中的参数ε。实验中，各案例区ε取值为 5%。因案例 GY 和 GY2 中空间单元属性差异很大，分区数量不能过多，否则难以满足均等约束。

实验计算环境为：HP 桌面计算机，配置 Intel Core I7-6700 CPU 3.40-GHz 和 8GB 内存，Windows 10 操作系统，安装有 Python 2.7、ArcGIS 10.4 和 IBM CPLEX 12.6 优化器。在 ArcGIS 中准备案例数据，包括空间单元及其属性 p_i、单元邻接关系 c_{ij} 和距离 d_{ij}(以单元中心点为基准)。本文算法使用 Python 程序设计语言编程实现。为提升计算速度，算法在 PyPy7 (https://www.pypy.org) 环境中运行。

3.2 计算结果

针对给个案例区域和分区数量使用两种方法求解。第一，按照数学表达式(1)-(11)构建案例模型，使用 CPLEX12.6 优化器进行求解。因区域 GY2 和 HN 规模过大，仅完成了区域 HD、GY 和 ZY 案例的模型求

解。第二，使用 2.2 节算法求解。算法参数设置如下：*psize*=10，*strength*=5%，*mloops*=100，SPP 模型求解时间限制 *t*=100s。为验证算法有效性，每个区域的每个分区均计算 10 次，统计其平均目标值和标准差，并与模型计算提供的目标值下界作比较。

案例区 HD 分区结果见表 2，分区示意图见图 2。表中，LB 和 UB 分别表示目标函数(1)的下界和上界，MIPGap 为 CPLEX 求解结果与下界之差异，Opt.表示计算结果为最优解。因该区域规模较小，模型计算较快获得最优解，混合算法也能够快速获得最优解。

表 2 区域 HD 分区结果

Table 2 Solution results from area HD

分区数量	参数 ε	CPLEX				混合算法		
		LB	UB	MIPGap	Time/s	Gap	Dev	Time/s
2	5%	0.099	0.099	Opt.	0.99	0.00%	0.00%	0.32
3	5%	0.084	0.084	Opt.	1.20	0.00%	0.00%	0.44
4	5%	0.071	0.071	Opt.	1.14	0.00%	0.00%	0.52
5	5%	0.060	0.060	Opt.	1.31	0.00%	0.00%	0.62

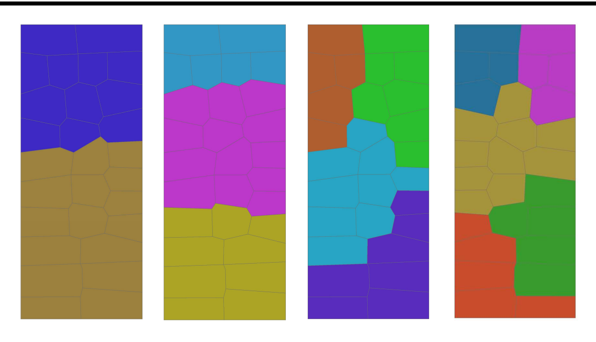


图 2 区域 HD 分区示意图

Fig. 2 Districting maps of area HD

案例区 GY 分区结果见表 3，部分分区示意图见图 3。可以看出，CPLEX 优化器能够求解空间单元数量为 297 的分区案例。文献[5]中求解类似问题的模型，难以求解 36 空间单元的分区案例。本文混合算法分区数量为 2、3、7 和 10 时求解质量高，能够获得最优解或极其接近最优解；分区数量为 5、6 和 8 时，能够获得近似最优解；分区数量为 4 时，求解质量略差。与模型精确求解相比，混合算法效率非常高，仅需 6~10 秒。

表 3 区域 GY 分区结果

Table 3 Solution results from area GY

分区数量	参数 ε	CPLEX				混合算法		
		LB	UB	MIPGap	Time/s	Gap	Dev	Time/s
2	5%	8.149	8.149	Opt.	6,098.57	0.00%	0.00%	6.20
3	5%	6.085	6.085	Opt.	3,990.47	0.04%	0.06%	6.18
4	5%	5.135	5.135	Opt.	11,145.66	1.19%	0.03%	7.51
5	5%	4.616	4.616	Opt.	7,588.53	0.16%	0.07%	9.92
6	5%	4.185	4.185	Opt.	11,303.26	0.24%	0.37%	6.67
7	5%	3.751	3.751	Opt.	3,576.36	0.00%	0.00%	6.17
8	5%	3.529	3.529	Opt.	5,390.92	0.22%	0.09%	6.92
10	5%	3.120	3.120	Opt.	4,823.14	0.00%	0.01%	7.18

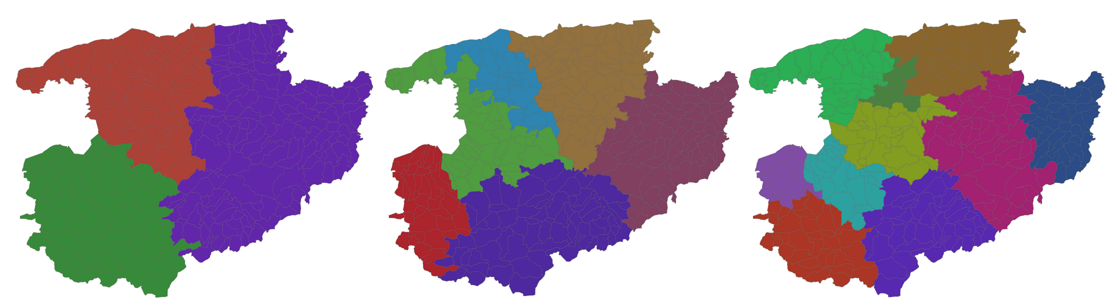


图 3 区域 GY 分区示意图

Fig. 3 Districting maps of area GY

案例区 ZY 分区结果见表 4，部分分区结果示意图见图 4。可以看出，所有案例模型均获得最优解，且混合算法结果逼近了最优解，仅分区数量为 20 时，目标值与最优解相差 0.29%。另外，混合算法稳定性较好，10 次计算结果的标准差较小。

表 4 区域 ZY 分区结果

Table 4 Solution results from area ZY

分区数量	参数 ε	CPLEX				混合算法		
		LB	UB	MIPGap	Time/s	Gap	Dev	Time/s
2	5%	0.979	0.979	Opt.	3,480.72	0.00%	0.00%	5.24
3	5%	0.831	0.831	Opt.	3,796.54	0.00%	0.00%	5.75
4	5%	0.717	0.717	Opt.	4,292.85	0.01%	0.04%	5.84
5	5%	0.624	0.624	Opt.	4,758.05	0.05%	0.03%	5.80
6	5%	0.562	0.562	Opt.	5,811.00	0.08%	0.02%	6.04
7	5%	0.513	0.513	Opt.	5,927.70	0.01%	0.01%	6.00
8	5%	0.477	0.477	Opt.	4,544.64	0.02%	0.04%	7.32
10	5%	0.425	0.425	Opt.	4,973.77	0.02%	0.01%	7.03
15	5%	0.351	0.351	Opt.	4,973.77	0.17%	0.10%	9.84
20	5%	0.297	0.297	Opt.	5,397.47	0.29%	0.19%	15.54

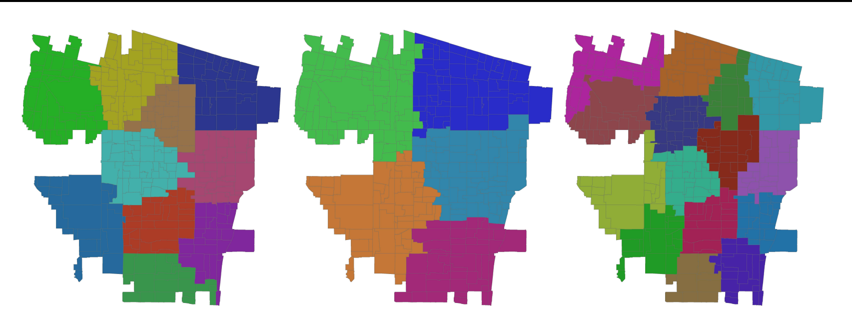


图 4 区域 ZY 分区示意图

Fig. 4 Districting maps of area ZY

案例区 GY2 分区结果见表 5，部分分区示意图见图 5。案例区 GY2 和 HN 空间单元数量较大，CPLEX 优化器求解数学模型时内存溢出。混合算法求解结果稳定性较好，标准差处于 0.03%~0.11%之间。该区域有 1276 个空间单元，能够在 2~5 分钟内完成分区计算。

表 5 区域 GY2 分区结果

Table 5 Solution results from area GY2

分区数量	参数ε	最小	平均	最大	标准差	Time/s
5	5%	4.439	4.442	4.443	0.03%	115.78
10	5%	3.108	3.109	3.110	0.03%	123.96
15	5%	2.544	2.548	2.551	0.08%	268.82
20	5%	2.174	2.177	2.181	0.11%	165.05

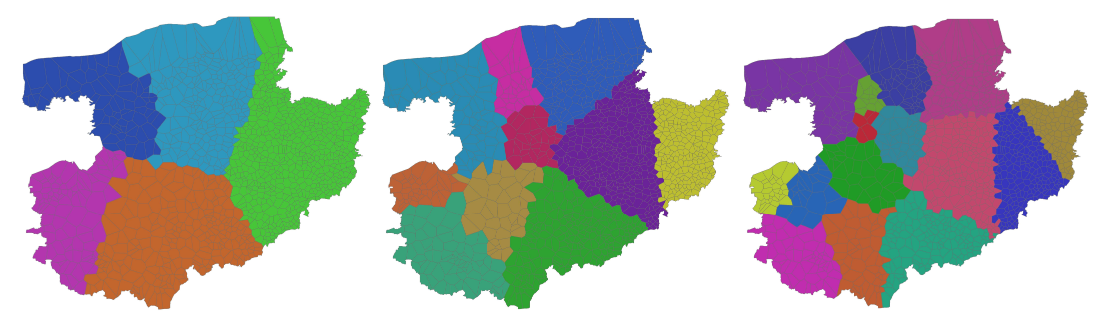


图 5 区域 GY2 分区示意图

Fig. 5 Districting maps of area GY2

案例区 HN 分区结果见表 6，部分分区示意图见图 6。混合算法求解结果稳定性较好，标准差处于 0.00%~0.18%之间。另外，算法需要 5~9 分钟完成分区。

表 6 区域 HN 分区结果

Table 6 Solution results from area HN

分区数量	参数ε	最小	平均	最大	标准差	Time/s
5	5%	69.762	69.766	69.769	0.00%	302.46
10	5%	46.548	46.560	46.586	0.02%	330.24
15	5%	37.562	37.571	37.581	0.02%	546.56
20	5%	31.781	31.795	31.810	0.03%	404.85
25	5%	28.291	28.307	28.327	0.04%	539.99
30	5%	25.854	25.864	25.879	0.03%	340.52
35	5%	23.764	23.798	23.832	0.09%	323.62
40	5%	22.244	22.296	22.333	0.13%	498.67
45	5%	20.760	20.802	20.860	0.18%	388.39
50	5%	19.541	19.590	19.612	0.12%	410.77

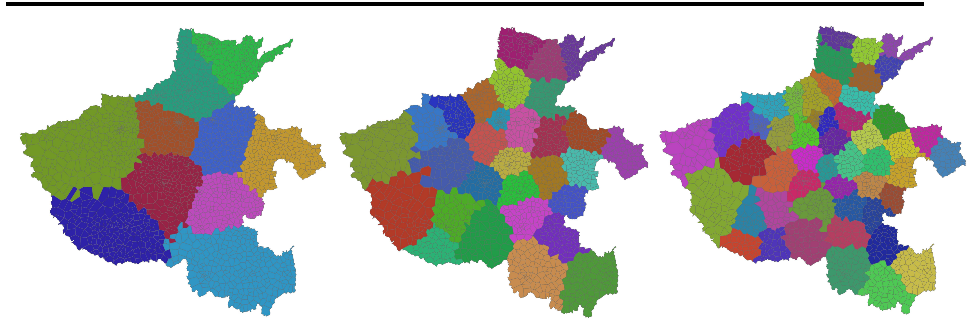


图 6 区域 HN 分区示意图

Fig. 6 Districting maps of area HN

3.3 应急服务应用

除应用于经典的选区、销售区和巡逻区划分外，均等分区还能够广泛应用于应急管理、城市管理责任区划分、区域公共服务等。假定

在新冠病毒流行期间，疾控部门启动某区域全员核酸检测，派出若干小组完成测试。均等分区结果可满足检测基本要求：每个组任务均等，能快速完成检测任务；每个服务站服务的居民尽可能集中，从而使居民到达服务站步行距离最短；避免聚集，即居民仅和近邻居民一起排队，避免与居住较远的居民聚集在一起；方便管理，每个服务点的服务区域空间连续，便于发布通知、上门服务等。案例 HD 适用于居民小区全员核酸检测。在图 2 中各分区中心位置或中心附近设置服务点，即可较好地完成服务点与服务区布局规划。

均等分区也适用于应急物资配送。假定在新冠病毒流行期间，某区域实施封闭管理。为保障医药或生活物资供应，应急指挥部门根据本区域参与物资供应商家及自愿者情况，将其划分为若干物资配送小组，每个小组负责一个区域的物资供应。均等分区可以保证每个小组任务量接近，分区形状紧凑能够减少交通时间。例如，区域 ZY 和 GY 分区可作为城市和农村地区物资供应服务区划分。因计算速度快，指挥部门可以根据参与商家和自愿者的动态变化不断调整服务区，优化物资配送服务。

4 讨论与结论

针对均等分区问题，本文构造了一个混合整型线性规划模型，并设计了一个基于迭代局部搜索算法的混合算法。选择 5 个区域对算法进行测试，结果发现：①本文模型能够求解空间单元数量达到 324 的案例；②本文设计的混合算法优化性能优异，鲁棒性强，计算效率较高。

本文在三个方面区别于现有模型：①通过给定分区中心单元，使分区连续约束条件得到简化。与现有模型相比，该模型不需要引入汇流单元决策变量[4][5]。文献[4]均质分区问题模型仅能够求解空间单元数为 16~49 的部分案例，文献[5]政治分区问题仅能够求解空间单元数为 25~36 的部分案例，而本文模型可用于求解单元数量达 324 的较大规模案例。②通过给定分区中心单元，易于建立紧凑度目标函数。分区形状紧凑度的定义方式很多，本文使用加权旅行距离作为紧凑度目标，物理意义明确，易于构造线性函数，也避免了文献[4]引入额外的虚拟决策变量。③本文模型本质上是容量约束 Location-Allocation 模型和空间连续约束的组合。模型求解结果提供中心区位，从而使均等分区问题能够应用于多种服务规划，特别是对于服务设施位置条件要求不高的问题。例如，新冠疫情应急管理中，可以基于均等分区进行医药配送、生活物资配送和核酸检查等服务规划，较好地满足规划要求。

本文设计的混合算法是对 ILS 算法的扩展：基于群解搜索、VND 搜索及 SPP 模型改进。虽然案例测试表明算法性能优异，但仍需进一步研究。一是构造更多的案例对算法进行充分测试，二是进行算法分析，引入学习机制，自适应地调整算法参数。

参考文献：

[1] Kalcsics J. Districting Problems[M]//Laporte G, Nickel S and da Gama F S(eds), Location Science, Springer, 2015: 595-622.

[2] Ricca F, Scozzari A, Simeone B, et al. Political districting: from classical models to recent approaches[J]. A Quarterly Journal of Operations Research, 2011, 9(3): 223-254.

[3] Keane M C. The size of the region-building problem[J]. Environment and Planning A, 1975, 7(5): 575-577.

[4] Duque J C, Church R L, Middleton R S, et al. The p-Regions problem [J]. Geographical Analysis, 2011, 43(1): 104-126.

[5] Plane, D. A., Tong, D. and Lei T. Inter-person Separation: A New Objective Standard for Evaluating the Spatial Fairness of Political Redistricting Plans [J]. Geographical Analysis, 2019, 51: 251–279

[6] Hess S W, Weaver J B, Siegfeldt H J, et al. Nonpartisan Political Redistricting by Computer[J]. Operations Research, 1965, 13(6): 998-1006.

[7] Hojati M. Optimal political districting[J]. Computers & Operations Research, 1996, 23(12): 1147-1161.

[8] George J A, Lamar B W, Wallace C A. Political district determination using large-scale network optimization[J]. Socio-Economic Planning

Sciences, 1997, 31(1):11-28.
[9] Nemoto T, Hotta K. Modelling and solution of the problem of optimal electoral districting. Communications of Operations Research of Japan, 2003, 48(4): 300–306. (http://www.orsj.or.jp/~archive/pdf/bul/Vol.4804 30 0.pdf , in Japanese)
[10] Li Z, Wang R, Wang Y. A Quadratic programming model for political districting problem[C]//The First International Symposium on Optimization and Systems Biology (OSB'07) Beijing, China, 2007: 427–435
[11] Garfinkel R, Nemhauser G L. Optimal Political Districting by Implicit Enumeration Techniques[J]. Management Science, 1970, 16(8): 495-508.
[12] Nygreen B. European assembly constituencies for Wales - comparing of methods for solving a political districting problem[J]. Mathematical Programming, 1988, 42(1): 159–169.
[13] Mehrotra A, Johnson E L, Nemhauser G L. An Optimization Based Heuristic for Political Districting[J]. Management Science, 1998, 44(8):1100-1114.
[14] Damico S J, Wang S, Batta R, et al. A simulated annealing approach to police district design[J]. Computers & Operations Research, 2002, 29(6): 667-684.
[15] Bozkaya B, Erkut E, Laporte G, et al. A tabu search heuristic and adaptive memory procedure for political districting[J]. European Journal of Operational Research, 2003, 144(1): 12-26.
[16] Ricca F, Simeone B. Local search algorithms for political districting[J]. European Journal of Operational Research, 2008, 189(3): 1409-1426.
[17] Chou C. A Knowledge-based Evolution algorithm approach to political districting problem[J]. Computer Physics Communications, 2011, 182(1):209-212.
[18] Altman M, Mcdonald M P. BARD: Better automated redistricting[J]. Journal of Statistical Software, 2011, 42(i04):1-28.
[19] Guo and Jin,2011 Guo D, Jin H. iRedistrict: Geovisual analytics for redistricting optimization.[J]. Journal of Visual Languages & Computing, 2011, 22(4):279-289.
[20] Forman S L, Yue Y. Congressional districting using a tsp-based genetic algorithm[J]. Lecture Notes in Computer Science, 2003, 2724:2072-2083.
[21] Wei B C, Chai W Y. A multiobjective hybrid metaheuristic approach for gis-based spatial zoning model[J]. Journal of Mathematical Modelling & Algorithms, 2004, 3(3):245-261.
[22] Bacao F, Lobo V, Painho M, et al. Applying genetic algorithms to zone design[J]. Soft Computing, 2005, 9(5): 341-348.
[23] Chou C, Chu Y, Li S, et al. Evolutionary strategy for political districting problem using genetic algorithm[C]. International Conference on Conceptual Structures, 2007.
[24] Salazar-Aguilar M A, Rios-Mercado R Z, Gonzalez-Velarde J L, et al. multiobjective scatter search for a commercial territory design problem[J]. Annals of Operations Research, 2012, 199(1): 343-360.
[25] Rincón-García E A, Gutiérrez-Andrade M Á, De-Los-Cobos-Silva S G, et al. ABC, A Viable Algorithm for the Political Districting problem[M]//Scientific Methods for the Treatment of Uncertainty in Social Sciences. Springer International Publishing, 2015:269-278.
[26] Garcia E A, Deloscobossilva S G, Ponsich A, et al. A system for political districting in the state of mexico[C]. Mexican International Conference on Artificial Intelligence, 2015.
[27] Lei H, Wang R, Laporte G. Solving a multi-objective dynamic stochastic districting and routing problem with a co-evolutionary algorithm[J]. Computers & Operations Research, 2016, 67: 12–24.
[28] Liu Y, Choc W K, Wang S. PEAR: a massively parallel evolutionary computation approach for political redistricting optimization and analysis [J]. Swarm and Evolutionary Computation, 2016, 30: 78–92.
[29] Vickrey W. On the Prevention of Gerrymandering[J]. Political Science Quarterly, 1961,76(1): 105-110.
[30] Cirincione C, Darling T A, Orourke T G, et al.

Assessing South Carolina's 1990s congressional districting[J]. Political Geography, 2000, 19(2): 189-211.

[31] Kalcsics J, Nickel S, Schroder M, et al. Towards a unified territorial design approach: Applications, algorithms and GIS integration[J]. Top, 2005, 13(1): 1-56.

[32] Hor M, Peng Y. Improving electoral districting through geographic information systems[C]. International Conference on Computer Science and Intelligent Communication, 2015.

[33] Ricca F, Scozzari A, Simeone B. Weighted Voronoi region algorithms for political districting[J]. Mathematical & Computer Modelling, 2007, 48(9-10):1468-1477.

[34] Shirabe T. Districting modeling with exact contiguity constraints [J]. Environment and Planning B: Planning and Design, 2009, 36(6): 1053-1066.

[35] Bucarey V, Ordóñez F, Bassaletti E. Shape and balance in police districting[M]//Applications of Location Analysis. Springer International Publishing, 2015: 329-347.

[36] Fryer R G, Holden R. Measuring the compactness of political districting plans[J]. The Journal of Law and Economics, 2011, 54(3): 493-535.

[37] Kong, Y., Zhu, Y., and Wang, Y. A center-based modeling approach to solve the districting problem, International Journal of Geographical Information Science, 2019, 33(2), 368-384.

[38] Lourenço, H.R., Martin, O. and Stützle, T. Iterated Local Search: Framework and Applications [M]// Gendreau, M. and Potvin, J.Y., eds. Handbook of Metaheuristics, 2nd. Edition. New York: Springer, 2010, 363-397.

[39] Kong, Y., Zhu, Y. and Wang, Y. A hybrid metaheuristic algorithm for the school districting problem. Acta Geographica Sinica, 2017, 72(2): 256-268. (孔云峰,朱艳芳,王玉璟. 学校分区问题混合元启发算法研究[J]. 地理学报, 2017, (02): 256-268.)

The Equal Districting Problem: Model, Algorithm and Applications

KONG Yunfeng[1]

1 Key Laboratory of Geospatial Technology for the Middle and Lower Yellow River Regions, Ministry of Education, Henan University,

Abstract: Districting problems have been widely applied in geography, economics, environmental science, politics, business, public service and many other areas. The equal districting problem (EDP) arises in applications such as political redistricting, police patrol area delineation, sales territory design and some service area design. The important criteria for these problems are district equality, contiguity and compactness. A mixed integer linear programming (MILP) model and a hybrid algorithm are proposed for the EDP. The hybrid algorithm is designed by extending iterative local search (ILS) algorithm with three schemes: population-based ILS, variable neighborhood descent (VND) local search, and set partitioning. The performance of the algorithm was tested on five areas. Experimentation showed that the instances could be solved effectively and efficiently. The potential applications of the EDP in emergency services are also discussed.

Key words: equal districting problem; mixed integer linear programming; hybrid algorithm; emergency service

Corresponding author: KONG Yunfeng, PhD, professor, specializes in spatial optimization, E-mail: yfkong@henu.edu.cn

Foundation Support: The National Natural Science Foundation of China, No. 41871307.